Within the loop 471, in step 472, the resource manager 121 creates (e.g., in the memory 112) a representation of the object 302 (the object being obtained from the tree traversal process) and includes in the representation the simple name of the object 304.

Next, the resource manager 121 performs steps 473 through 475 to determine if a home condition exists for the object 302.

Specifically, in step 473, the resource manager 121 determines if the object location of the object 302 in the object hierarchy 301 is the home 305 of the object. In other words, in step 473, the resource manager 121 examines the current tree location of the object 302 (from the tree traversal process) and compares this location value against the home of the object 305 as defined within the object 302. If they are the same, processing proceeds to step 476. However, if the location within the tree is not the same as the home 305 of the object 302, then this means that a representation of the object is going to displayed outside of its home context and processing proceeds to step 474.

In step 474, the resource manager 121 determines if the simple name of the object 304 is unique in the non-home object context in which this representation is to be displayed. In other words, the resource manager 121 determines if the simple name is unique in the hierarchy placement or location in relation to other objects in that same context or hierarchy location placement. This may be done, for example, by comparing the simple name 304 with the simple names 304 of all other objects 302 at that same location in the object hierarchy 301. If the simple name 304 is not unique, processing proceeds to step 475.

In step 475, the resource manager 121 appends the home of the object 305 to the representation of the object (i.e., in memory, awaiting to be displayed) which will be displayed within the graphical user interface 150. In this manner, when the resource manager 121 is to display a representation of an object 302 out of its home context and its simple name 304 is not unique, the representation will include both the simple name of the object 304 as well as the home of the object 305.

If after processing either step 473 and possibly step 474 and then also possibly step 475, processing proceeds to step 476.

In step 476, the resource manager 121 determines if the home of the object 305 is unique within the entire object hierarchy 301. In other words, in step 476, the resource manager 121 determines if there is another object 302 in the resource hierarchy 301 that is considered a home to another object and that has the same simple name as the name of the home 305 of the object 302 being processed according to the step in Figure 7. For example, a data storage system represented by a home object 302 having a simple name which is the same as a simple name of a host or of a network device represented by another object 302 creates a situation in which two resources are both homes to other resources and each has the same simple name 304. In such circumstances, since each resource is a home to other resources and has the same simple name, then a home exists that is not unique for objects below those resources in the object hierarchy 301 in the sense that two homes for other resources exist with the same simple name. In such instances, the processing of the resource manager 121 proceeds to step 477 for application of a suffix to the home of the object 305 being processed in the loop 471. As noted above, the suffix (when displayed on the graphical user interface) indicates to the user 108 that there is at least two home resources having the same name.

In step 477, the resource manager 121 appends a suffix to the home of the object 305 and includes the home of the object 305 in the representation of the object being created according to the processing of the loop 471. The suffix can be any character string, symbol or other designation that can indicate to a user that this home is not unique in the computing system environment 100.

After processing steps 476 and possibly 477 (if the suffix is required), processing proceeds to step 478 at which point the resource manager 121 displays a representation of the object 302 in a graphical user interface 150 in a location corresponding to the object location within the object hierarchy 301. In this manner, the object 302 is visually represented on the graphical user interface 150.

Processing then returns to step 472 at which point the resource manager 121 traverses the tree to process and display a representation of the next object 302 in the object hierarchy 301 for display within the graphical user interface 150. Using these techniques, embodiments of the invention are able to properly represent objects with a

graphical user interface using the unique naming and grouping techniques explained herein.

Those skilled in the art will understand that there can be many other variations made to the operations of the embodiments explained above while still achieving the same objectives of the invention. Such variations are intended to be covered by the scope of this invention. As such, the foregoing description of embodiments of the invention are not intended to be limiting. Rather, any limitations to embodiments of the invention are presented in the following claims.